StrCatChainW

The result of StrCatChainW() is not guaranteed to be null terminated.

Sean Barnum, Cigital, Inc. [vita[1]]

2007-04-17

Part "Original Cigital Coding Rule in XML"

Mime-type: text/xml, size: 4053 bytes

<table>
<tr><td><b>Attack Category</b></td><td><ul><li>Denial of Service</li><li>Malicious Input</li></ul></td></tr>
<tr><td><b>Vulnerability Category</b></td><td><ul><li>Buffer Overflow</li><li>Unconditional</li><li>No Null Termination</li></ul></td></tr>
<tr><td><b>Software Context</b></td><td><ul><li>String Management</li></ul></td></tr>
<tr><td><b>Location</b></td><td><ul><li>shlwapi.h</li></ul></td></tr>
<tr><td><b>Description</b></td><td><p>The result of StrCatChainW() is not guaranteed to be null terminated.</p><p>This function StrCatChainW() concatenates two Unicode strings. It is good for repeated calls in a loop because after appending the strings it returns the offset to begin the next concatenation. It takes a buffer size and does a good job at not overrunning the buffer. However, The final string is NOT guaranteed to be null terminated.</p></td></tr>
<tr><td><b>APIs</b></td><td><table><tr><th>Function Name</th><th>Comments</th></tr><tr><td>StrCatChainW</td><td></td></tr></table></td></tr>
<tr><td><b>Method of Attack</b></td><td><p>In boundary cases where the concatenated strings exactly fit into the allocated buffer, the string might not be null terminated. This can lead to access violations the next time the string is accessed.</p><p>An attacker can exploit this function to implement a denial of service (DoS) or buffer overflow (BO) attack. BO may lead to a DoS attack against the application if an access violation occurs. In the worst case, a BO may allow an attacker to inject executable code into your process, especially if the destination string is a stack-based buffer.</p></td></tr>
<tr><td><b>Exception Criteria</b></td><td></td></tr>
<tr><td><b>Solutions</b></td><td><table><tr><th>Solution Applicability</th><th>Solution Description</th><th>Solution Efficacy</th></tr></table></td></tr>
</table>

[1] http://buildsecurityin.us-cert.gov/bsi-rules/35-BSI.html (Barnum, Sean)

<table>
<tr><td></td><td><table><tr><td>When StrCatChainW() is used.</td><td>With suitable care, StrCatChainW() can be used safely. However, it is probably better to use one of the following alternatives: StringCbCatEx or StringCchCatEx.</td><td>Effective, but still requires appropriate usage.</td></tr></table></td></tr>
<tr><td><b>Signature Details</b></td><td>DWORD StrCatChainW(
LPWSTR pszDst,
DWORD cchDst,
DWORD ichAt,
LPCWSTR pszSrc
);</td></tr>
<tr><td><b>Examples of Incorrect Code</b></td><td><pre>for (i=0; i<6; i++) {
ichAt = StrCatChainW(pszDst,
cchBufSize, ichAt, L"abc");
ichAt = StrCatChainW(pszDst,
cchBufSize, ichAt, L"xyz");
}</pre></td></tr>
<tr><td><b>Examples of Corrected Code</b></td><td><pre>LPTSTR pszAppend = pszDst; //
jumpstart at beginning, update
after each strcat
for (i=0; i<6; i++) {
StringCbCatExW(pszAppend,
cchBufSize, L"abc", &pszAppend,
NULL, STRSAFE_IGNORE_NULLS);
StringCbCatExW(pszAppend,
cchBufSize, L"xyz", &pszAppend,
NULL, STRSAFE_IGNORE_NULLS);
}</pre></td></tr>
<tr><td><b>Source Reference</b></td><td>• http://msdn.microsoft.com/library/default.asp?url=/library/en-us/shellcc/platform/shell/reference/shlwapi/string/strcatchainw.asp[2]</td></tr>
<tr><td><b>Recommended Resource</b></td><td></td></tr>
<tr><td><b>Discriminant Set</b></td><td><table><tr><td><b>Operating System</b></td><td>• Windows</td></tr><tr><td><b>Languages</b></td><td>• C
• C++</td></tr></table></td></tr>
</table>

Cigital, Inc. Copyright

The Build Security In (BSI) portal is sponsored by the U.S. Department of Homeland Security (DHS), National Cyber Security Division. The Software Engineering Institute (SEI) develops and operates BSI. DHS funding supports the publishing of all site content.

[1] mailto:copyright@cigital.com